

Ezoteryczne języki programowania - zadania kwalifikacyjne

Rozwiązania zadań wysyłajcie na adres `lisklusownik+www14@gmail.com`.

Zadanie 1 (10 punktów)

Zaproponuj implementację interpretera do języka Polynomial. Użyty język może być dowolny.

Zasady oceniania:

- jeśli program nie wykonuje poprawnie programów, może dostać maksymalnie 4 punkty.
- punkty przyznawane są zależnie od tego jak mi się spodoba struktura kodu
- przewiduję dodatkowe punkty, jeśli interpreter będzie obsługiwał więcej opcji niż minimum (i będą to opcje użyteczne, mając na uwadze programowanie w Polynomialu).

Uwagi

Przykładowe wejście:

$$f(x) = x^4 - 27x^3 + 59x^2 - 243x + 450$$

W szczególności,

- wejście jest łańcuchem znaków
- na początku zawsze jest $f(x) =$
- nie ma znaku mnożenia pomiędzy współczynnikiem, a symbolem x
- jeśli x^n ma współczynnik 0, to można je pominąć (czego akurat nie widać na przykładzie, ale niemniej jest to zasadą)
- podobnie z x^0
- tak jak na przykładzie, możemy oczekiwać spacji tam, gdzie jest to sensowne

- jeśli $a + ib$ jest zerem wielomianu, oraz $b \neq 0$, to można założyć, że $|a| < 256$.

Zadanie 2 (10 punktów)

Stwórz jak najkrótszy program realizujący problem:

Dana jest liczba całkowita $n > 1$.

1. Policz sumę liczb $n, n-1, n-2, \dots, 3, 2, 1$
2. Policz iloczyn cyfr powyższej liczby
3. Policz sumę cyfr powyższej liczby
4. Wykonuj kroki 2. oraz 3. dopóki nie zostanie jedna cyfra. Ta cyfra jest wynikiem.

Pierwsze dwadzieścia wartości (poczynając od $n=2$):

3, 6, 0, 5, 2, 7, 9, 2, 7, 9, 1, 9, 0, 0, 9, 6, 7, 0, 0, 6

Zasady:

- Program powinien być napisany w Pythonie 3 — robię to aby móc porównywać programy napisane przez różnych aplikujących
- Program powinien radzić sobie z liczbami mniejszymi niż 1000000, nie trzeba się przejmować co się stanie, jeśli wejście jest większe.
- Zarówno pełen program, jak i pojedyncza funkcja są dopuszczalne. Jeśli zdecydował(a|e)ś się na funkcję, zwróć wynik, a nie drukuj.

Ocenianie

Długością programu jest ilość bajtów, z których się składa. Przykładowo, długość poniższego programu, to 22 (druga linijka zawiera 4 spacje wcięcia):

```
def f(x):
    print(x)
```

Niech długością programu osoby i będzie $C[i]$. Wówczas wynik osoby i jest równy $10 * \min(C) / C[i]$, chyba że program nie jest poprawny.

Przykłady:

n	wyjście
1234	9
3005	3

5007	5
9854	8
75849	8
100000	0

Wskazówki:

Programy można testować na stronie Try it Online, przy okazji strona ta podaje ilość bajtów, które składają się na rozwiązanie. Poza tym, warto rzucić okiem na:

- [Tips for golfing in <all languages>](#)
- [Tips for golfing in Python](#)